AD-A046 316 FLORIDA UNIV GAINESVILLE DEPT OF INDUSTRIAL AND SYS--ETC F/G 12/1
CIRCULAR 1'S AND CYCLIC STAFFING.(U)
SEP 77 J J BARTHOLDI, J B ORLIN, H D RATLIFF N00014-76-C-0096
UNCLASSIFIED RR-77-11 NL

1 OF 1
AD
A046 316

END
DATE
FILMED
12-77
DDC

1

B.S.

RESEARCH REPORT

Industrial & Systems
Engineering Department
University of Florida
Gainesville, FL. 32611

CIRCULAR 1'S AND CYCLIC STAFFING

Research Report 77-11

by

John J. Bartholdi III
James B. Orlin*
H. Donald Ratliff

September, 1977

Department of Industrial and Systems Engineering
University of Florida
Gainesville, Florida 32611

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

This research was supported in part by the Office of Naval Research under contract number N00014-76-C-0096 and the Army Research Office under contract number DAHC04-75-G-0150 at the University of Florida and, at Stanford University, by the Office of Naval Research under contract number N00014-75-C-0493 and National Science Foundation under contract ENG 76-12266.

THE FINDINGS OF THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY OR NAVY POSITION, UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

*Stanford University

See 1473

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

(14) 1. REPORT NUMBER: RR-77-11

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER: (9) Research rept.,

4. TITLE (and Subtitle): (6) Circular 1's and Cyclic Staffing.

5. TYPE OF REPORT & PERIOD COVERED: Technical

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): (10) John J./Bartholdi, III, James B./Orlin, H. Donald/Ratliff

8. CONTRACT OR GRANT NUMBER(s): (15) N00014-76-C-0096, N00014-75-C-0493, DAHC04-75-G-0150, ENG 76-12266.

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
U.S. Army Res. Off., P.O. Box 12211, Triangle Park, NC 27709
Office of Naval Research, Arlington, VA

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: (16) 20061102A14D Rsch in & Appl of Applied Math.

11. XXXXXXXXXXXXXXXXXXXXXXXXXX CONTROLLING OFFICE NAME AND ADDRESS:
National Science Foundation
Engineering Division
Washington, D.C.

12. REPORT DATE: (11) Sep[redacted]77

13. NUMBER OF PAGES: 29 (12) 30p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

N/A

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Networks
Integer Programming
Circular 1's
Scheduling
Staffing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A commonly encountered integer linear program, basic to cyclic staffing and scheduling, has a constraint matrix possessing the property of "circular 1's in columns." In general, such a matrix is not unimodular, balanced, or perfect. Nevertheless, many such problems may be efficiently solved for integer answers. A change of variable transforms them to comfortably

next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

404 399

20.

finite and reassuringly predictable series of minimum cost network flow problems.

Abstract

A commonly encountered integer linear program, basic to cyclic staffing and scheduling, has a constraint matrix possessing the property of "circular 1's in columns." In general, such a matrix is not unimodular, balanced, or perfect. Nevertheless, many such problems may be efficiently solved for integer answers. A change of variable transforms them to comfortably finite and reassuringly predictable series of minimum cost network flow problems.

Table of Contents

CIRCULAR 1's AND CYCLIC STAFFING

1. Two Fundamental Staffing Models

Consider the integer linear program

$$\begin{aligned} \min\ & \bar{c}\bar{x} \\ \text{s.t. } & A\bar{x} \geq \bar{b} \\ & \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{1.1}$$

where, throughout the paper, $\bar{b}$ and $\bar{c}$ are vectors with all entries integer and A is an m x n matrix with all entries 0 or 1. Without loss of generality, we may assume $\bar{b}, \bar{c} \geq 0$.

To represent continuous workshifts in linear time, a common staffing model has A possess the property of consecutive 1's in columns (e.g., Veinott and Wagner [19]). Such matrices are happily met since they are known to be totally unimodular; moreover, for such matrices, problem (1.1) is equivalent under linear transformation to the minimum-cost network flow problem

$$\begin{aligned} \min\ & \bar{c}\bar{x} \\ \text{s.t. } & [TA, -T]\bar{x} = T\bar{b} \\ & \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{1.2}$$

where T is the m x m matrix

$$T = \begin{bmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & \ddots & & \\ & & \ddots & 1 & \\ & & & -1 & 1 \end{bmatrix} \tag{1.3}$$

and where "equivalence" means here that $\bar{x}$ solves (1.1) if and only if $\bar{x}$ solves (1.2) [11,12]. Transforming (1.1) by T to reveal its network structure corresponds to successively row-reducing the constraints of (1.1) [19]. Since the minimum cost network flow algorithm is formally efficient [8], we may consider (1.1) to be efficiently solvable in its guise (1.2).

The second basic staffing model represents continuous workshifts in cyclical time [3]. For this model the matrix A possesses the property of circular 1's in columns [18], as for instance in Example 1.1, where the strings of 1's may be imagined to wrap around the matrix. Such matrices are in general neither unimodular, balanced, nor perfect [16]. Indeed they are notorious for the fractional extreme points which they induce in (1.1) [13].

A special n x n circular 1's matrix has in each column a band of k 1's permuted cyclically (see Examples 1.2 and 1.3). We will call these (k, n) matrices.

The most fundamental of the cyclic staffing models is given by

$$\begin{aligned} &\min \bar{1}\bar{x} \\ &\text{s.t. } A\bar{x} \geq \bar{b} \\ &\quad \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{1.4}$$

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Example 1.1: A matrix with "circular 1's in columns."

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Example 1.2: A (2, 3) matrix. Example 1.3: A (3, 5) matrix.

where A is a (k, n) matrix. The objective corresponds to minimizing the total workforce size necessary to meet period manpower requirements $\bar{b}$. For A a (5, 7) matrix, this problem was studied by Tibrewala, Philippe, and Browne [17] (also by various others [2,4,6,15]), where the (5, 7) matrix represents workshifts with two consecutive days off each week. They observe that their solution generalizes to A a (k, k + 2) constraint matrix. A rather more complex solution technique is proposed by Guha [10] for general (k, n) matrices. In this study we generalize problem (1.4) in two ways and offer considerably simpler and formally efficient solutions.

2. Transformations of Variables

Consider the problem

$$\begin{aligned} &\min \bar{c}\bar{x} \\ &\text{s.t. } A\bar{x} \geq \bar{b} \\ &\qquad \bar{x} \geq \bar{0} \end{aligned} \tag{2.1}$$

which we may write as

$$\begin{aligned} &\min \bar{c}\bar{x} \\ &\text{s.t. } \begin{bmatrix} A \\ I \end{bmatrix} \bar{x} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \end{aligned} \tag{2.2}$$

Now let T be a nonsingular matrix and consider the change of variables $\bar{x} = T\tilde{y}$. Since T is nonsingular, (2.2) is equivalent to

$$\min\ (\bar{c}T)\bar{y}$$

$$\text{s.t.} \begin{bmatrix} AT \\ T \end{bmatrix} \bar{y} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \qquad (2.3)$$

$$\bar{y} \text{ unrestricted}$$

in the sense that if $\bar{x}$ is feasible to (2.2), then $T^{-1}\bar{x}$ is feasible to (2.3), and if $\bar{y}$ is feasible to (2.3), then $T\bar{y}$ is feasible to (2.2). If in addition T is unimodular, then if $\bar{x}$ has all integer entries, $T^{-1}\bar{x}$ has all integer entries, and if $\bar{y}$ has all integer entries $T\bar{y}$ has all integer entries. Therefore,

Observation 2.1: For T nonsingular and unimodular the integer-constrained versions of problems (2.2) and (2.3) are equivalent in the sense that $\bar{x}$ solves (2.2) iff $\bar{y} = T^{-1}\bar{x}$ solves (2.3).

We will use this insight to construct equivalent integer programs wherein special, exploitable structure is displayed.

3. Almost a Network

A key idea of this paper is that under certain conditions, when A is a circular 1's matrix, problem (1.1) is "almost" a network flow problem. Problem (3.1) below, where A is a (3, 5) matrix, will provide a continuing illustration of this class of problems. Later we will observe that the ideas to follow generalize easily.

$$\min\ c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5$$

$$\text{s.t.} \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \bar{x} \geq \begin{bmatrix} \\ \\ \bar{b} \\ \\ \\ - \\ \\ \bar{0} \\ \\ \\ \end{bmatrix} \tag{3.1}$$

$$\bar{x} \text{ integer}$$

where the nonnegativity constraints are expressed by the lower portion of the matrix. Perform the change of variable given by $\bar{x} = T\bar{y}$ where T is defined in (1.3). Such a change of variable corresponds to successive column reduction of matrix A, and results in

$$\min\ (c_1-c_2)y_1 + (c_2-c_3)y_2 + (c_3-c_4)y_3 + (c_4-c_5)y_4 + c_5y_5$$

$$\text{s.t.} \begin{bmatrix} 1 & 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \bar{y} \geq \begin{bmatrix} \\ \\ \bar{b} \\ \\ \\ - \\ \\ \bar{0} \\ \\ \\ \end{bmatrix} \tag{3.2}$$

$$\bar{y} \text{ unrestricted but integer}$$

Moreover, since such a T is both nonsingular and unimodular, (3.2) is equivalent to (3.1) as an integer linear program. Thus solving (3.2) solves (3.1). We solve (3.2), on the strength of the following,

Observation 3.1: With the exception of the last column, that corresponding to y_5, problem (3.2) is the linear programming dual of a network flow problem.

That is, if we fix (temporarily) y_5, problem (3.2) may be written as

$$c_5y_5 + \min\ (c_1-c_2)y_1 + (c_2-c_3)y_2 + (c_3-c_4)y_3 + (c_4-c_5)y_4$$

$$\text{s.t.} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \geq \begin{bmatrix} b_1 - y_5 \\ b_2 - y_5 \\ b_3 \\ b_4 \\ b_5 - y_5 \\ \hline 0 \\ 0 \\ 0 \\ 0 \\ -y_5 \end{bmatrix} \qquad (3.3)$$

$$y_1,\ y_2,\ y_3,\ y_4 \text{ unrestricted but integer}$$

which is the linear programming dual of a network flow problem. The obvious idea is to fix values of y_5 over its allowable range and solve corresponding network flow problems until the best objective value is found. Let us refine and extend this idea.

4. Properly Compatible 1's

Following Tucker [18], we define a 0 - 1 matrix A to have properly compatible circular 1's in columns if and only if (i) the 1's in each column are circular, and (ii) for any two columns $\bar{a}_j$ and $\bar{a}_k$, if the first (in a cyclic sense) 1 in $\bar{a}_j$ preceeds that of $\bar{a}_k$, then the last (in a cyclic sense) 1 in $\bar{a}_k$ does not preceed that of $\bar{a}_j$. Roughly speaking, if a circular band starts later than another, it can end no earlier. The matrices of Example 4.1 illustrate properly compatible circular 1's in columns. For matrices with this property, a natural ordering of the columns suggests itself,

Column Ordering Algorithm

1. Order columns in groups, where group i consists of those $\bar{a}_j$ whose first (in a cyclic sense) 1 appears in row i. Then,

2. Within each group, order columns so that $\bar{a}_j$ preceeds $\bar{a}_k$ if the last (in a cyclic sense) 1 of $\bar{a}_j$ preceeds that of $\bar{a}_k$. The columns of Example 4.1 have been so ordered. Henceforth, we assume, without loss of generality, that a matrix with properly compatible 1's in columns has its columns ordered as above. Important for us shortly will be

Observation 4.1: A matrix with properly compatible circular 1's in columns has the property of circular 1's in rows.

Consider now the problem

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Example 4.1a: Properly compatible circular 1's in columns.

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Example 4.1b: A circular 1's matrix, the columns of which are not properly compatible.

$$\begin{aligned} &\min \bar{c}\bar{x} \\ &\text{s.t. } A\bar{x} \geq \bar{b} \\ &\qquad \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{4.1}$$

where A has properly compatible 1's in columns. Perform the change of variables given by $\bar{x} = T\bar{y}$, where T is the non-singular unimodular matrix defined in (1.3). Then, we have an equivalent integer linear program

$$\begin{aligned} &\min (\bar{c}T)\bar{y} \\ &\text{s.t. } \begin{bmatrix} AT \\ T \end{bmatrix} \bar{y} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \\ &\qquad \bar{y} \text{ unrestricted but integer} \end{aligned} \tag{4.2}$$

Since A has circular 1's in rows, each row $\bar{r}_i$ of A has either consecutive 1's or consecutive 0's [18]. Therefore each $\bar{r}_i$ is of the form

(i) $\bar{r}_i = (0,\ldots,0,\ 1,\ldots,1,\ 0,\ldots,0)$, or

(ii) $\bar{r}_i = (1,\ldots,1,\ 0,\ldots,0,\ 1,\ldots,1)$, or

(iii) $\bar{r}_i = (1,\ldots,1,\ 0,\ldots,0)$, or

(iv) $\bar{r}_i = (0,\ldots,0,\ 1,\ldots,1)$

But then each row $\bar{r}_iT$ of AT is of the form

(i) $\bar{r}_iT = (0,\ldots,0,\ -1,\ 0,\ldots,0,\ 1,\ 0,\ldots,0)$, or

(ii) $\bar{r}_iT = (0,\ldots,0,\ 1,\ 0,\ldots,0,\ -1,\ 0,\ldots,0,\ 1)$, or

(iii) $\bar{r}_iT = (0,\ldots,0,\ 1,\ 0,\ldots,0)$, or

(iv) $\bar{r}_iT = (0,\ldots,0,\ -1,\ 0,\ldots,0,\ 1)$, respectively.

Note that, excluding the n^{th} column, each row of $\begin{bmatrix} AT \\ T \end{bmatrix}$ has at

most one +1 and one -1, all other entries being 0.

For notational convenience, let us partition T into its n^{th} column and the remainder of the matrix: $T = [T_r, \bar{t}_n] = [T_r, \bar{e}_n]$, since the n^{th} column of T is $\bar{e}_n = (0,\ldots,0, 1)$. Similarly partition $\bar{y} = (\bar{y}_r, y_n)$. Then problem (4.2) may be rewritten as

$$\begin{aligned} \min\ & (\bar{c}T_r)\bar{y}_r + c_n y_n \\ \text{s.t.}\ & \left[\begin{array}{c|c} AT_r & \bar{a}_n \\ T_r & \bar{e}_n \end{array}\right] \begin{bmatrix} \bar{y}_r \\ y_n \end{bmatrix} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \\ & \bar{y}_r,\ y_n \text{ unrestricted but integer} \end{aligned} \tag{4.3}$$

Now we can formally state

Lemma 4.1: If for problem (4.1), A has properly compatible circular 1's in columns, then under the prescribed change of variables $\begin{bmatrix} AT_r \\ T_r \end{bmatrix}$ is the transpose of a network matrix.

That is, for fixed integral y_n, the resultant version of (4.3)

$$\begin{aligned} \min\ & (\bar{c}T_r)\bar{y}_r \\ \text{s.t.}\ & \begin{bmatrix} AT_r \\ T_r \end{bmatrix} \bar{y}_r \geq \begin{bmatrix} \bar{b} - \bar{a}_n y_n \\ \bar{0} - \bar{e}_n y_n \end{bmatrix} \\ & \bar{y}_r \text{ unrestricted but integer} \end{aligned} \tag{4.4}$$

is the linear programming dual of a network flow problem. Thus, problem (4.4) is efficiently solvable, at least through its dual. This suggests the idea of searching through the allowable values of y_n, solving a tractable subproblem (4.4) each time, to find a $(\bar{y}_r^*, y_n^*)$ which minimizes $(\bar{c}T_r)\bar{y}_r + c_n y_n$.

5. Stalking the Wild y_n

First we determine the allowable range of the integer y_n. Let y_n^* be a value of y_n in some optimal solution to (4.2), and let b_{max} be the largest entry in $\bar{b}$. Then

Lemma 5.1: $b_{max} \leq y_n^* \leq \bar{1}\bar{b}$ for some y_n^*

Proof: Since $\bar{y} = T^{-1}\bar{x}$, and

$$T^{-1} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ \vdots & & \ddots & \\ 1 & 1 & \cdots & 1 \end{bmatrix}$$

we have that $y_n = \bar{1}\bar{x}$. To show the lower bound, it is sufficient to observe that since $\bar{x} \geq 0$, $y_n = \bar{1}\bar{x} \geq \sum_j a_{ij}x_j \geq b_i \ \forall\ i$. Therefore, $y_n^* \geq b_{max}$.

To establish the upper bound, we may assume that at optimality every variable in (4.1) appears in some tight constraint, since otherwise that variable could be reduced, feasibility maintained, and the objective function not increased. Summing over the set S of tight constraints yields $\sum_{i\in S} \sum_j d_{ij}x_j^* = \sum_{i\in S} b_i$; but $\bar{1}\bar{b} \geq \sum_{i\in S} b_i = \sum_{i\in S}\sum_j d_{ij}x_j^* \geq \bar{1}\bar{x}^* = y_n^*$.

Q.E.D.

To remind us of its dependence on y_n, let the objective function of problem (4.4) be written as $(\bar{c}T_r)\bar{y}_r = z(y_n)$ and let the optimal value, for fixed y_n, be $z^*(y_n)$.

Lemma 5.2: $z^*(y_n)$ is convex in y_n over $b_{max} \leq y_n \leq \bar{1}\bar{b}$.

Proof: Since the constraint matrix of problem (4.4) is totally unimodular, the integral restrictions may be dropped. Then the desired conclusion follows from similar results

for continuous-valued linear programs (e.g., Geoffrion and Nauss [9]).

Q.E.D.

Lemma 5.3: The optimal function value $(\bar{c}T)\bar{y}$ of problem (4.2) is convex in y_n over $b_{max} \leq y_n \leq \bar{1}\bar{b}$.

Proof: Clearly $c_n y_n$ is convex in y_n; $z^*(y_n)$ is convex in y_n by Lemma 5.2, so since sums of convex functions are convex, $z^*(y_n) + c_n y_n$ is convex in y_n. But this is the optimal function value of problem (4.3) and therefore of problem (4.2)

Q.E.D.

6. A Solution Technique

Given the problem

$$\begin{aligned} &\min \bar{c}\bar{x} \\ &\text{s.t. } A\bar{x} \geq \bar{b} \\ &\qquad \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{6.1}$$

where A has properly compatible circular 1's in columns, the preceding results justify this solution procedure,

Step 0: Perform the change of variables

Let $\bar{x} = T\bar{y}$, where T is defined by (1.3) to form the equivalent problem

$$\begin{aligned} &\min \ (\bar{c}T)\bar{y} \\ &\text{s.t. } \begin{bmatrix} AT \\ T \end{bmatrix} \bar{y} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \\ &\qquad y \text{ unrestricted but integer} \end{aligned} \tag{6.2}$$

Step 1: Solve the equivalent problem, (6.2)

A. Note bounds on integer y_n^*: $b_{max} \le y_n^* \le \bar{1}\bar{b}$.

B. Minimize $z^*(y_n) + c_n y_n = (\bar{c}T)\bar{y}$ over this interval.

Since y_n is integer and $z^*(y_n) + c_n y_n$ is convex in y_n, an efficient technique such as Fibonacci search [21] may be used. Furthermore, for fixed y_n, $z^*(y_n)$ is readily calculated by solving

$$z^*(y_n) = \min\ (\bar{c}T_r)\bar{y}_r$$

$$\begin{bmatrix} AT_r \\ T_r \end{bmatrix} \bar{y}_r \ge \begin{bmatrix} \bar{b} - \bar{a}_n y_n \\ \bar{0} - \bar{e}_n y_n \end{bmatrix} \tag{6.3}$$

$$\bar{y}_r \text{ unrestricted but integer}$$

Since this is the dual of a minimum cost network flow problem, it is efficiently solvable. Let $(\bar{c}T)\bar{y}^* = \min_{y_n} z^*(y_n) + c_n y_n$ and let $(\bar{y}_r^*, y_n^*) = \bar{y}^*$ be the associated solution; then $\bar{y}^*$ solves (6.2) and $(\bar{c}T)\bar{y}^*$ is the optimal function value.

Step 2: Construct the optimal solution to (6.1) by the change of variables $\bar{x}^* = T\bar{y}^*$.

7. Efficiency of the Algorithm

This solution procedure works efficiently, even for pessimists, by the following worst-case analysis.

Step 0, the initial change of variables, requires no more than 0(mn) steps.

Step 1 requires the solution of (6.3) for fixed y_n. But the network flow algorithm solves (6.3) in a number of steps which is bounded above by a polynomial in the size of the encoding of the problem data [8]. We may take this

polynomial to be $p(m, n, \log_2 \bar{1}\bar{b}, \log_2 \bar{1}\bar{c}, y_n)$. But since $y_n^* \leq \bar{1}\bar{b}$, $\log_2 y_n^* \leq \log_2 \bar{1}\bar{b}$, so that we may consider the solution to (6.3) to require no more than $0(\hat{p}(m, n, \log_2 \bar{1}\bar{b}, \log_2 \bar{1}\bar{c}))$ for some polynomial $\hat{p}$. And since Fibonacci search requires that we consider no more than $0(\log_3 \bar{1}\bar{b})$ values of y_n, we may determine $\bar{y}^* = (y_r^*, y_n^*)$ is no more than $0(\log_3 \bar{1}\bar{b} \cdot \hat{p}(m, n, \log_2 \bar{1}\bar{b}, \log_2 \bar{1}\bar{c}))$ steps.

Step 2, the final change of variables, requires $0(n)$ steps.

Therefore the solution procedure solves (6.1) in at worst $0(mn + \log_3 \bar{1}\bar{b} \cdot \hat{p}(m, n, \log_2 \bar{1}\bar{b}, \log_2 \bar{1}\bar{c}))$ steps. Since this is polynomial in a binary encoding of the problem data [1], we have proven

Lemma 7.1: Problem (6.1) is solved by the solution technique with formal efficiency relative to a binary encoding of the problem data.

8. A Special Objective Function

For a special objective function, a wider class of problems may be solved and additional results discovered. Consider

$$\begin{aligned} &\min \ \bar{1}\bar{x} \\ &\text{s.t. } A\bar{x} \geq \bar{b} \\ &\qquad \bar{x} \geq \bar{0}, \text{ integer} \end{aligned} \tag{8.1}$$

where A displays the property of circular 1's in columns (not necessarily properly compatible).

We say that column $\bar{a}_j$ of A <u>dominates</u> column $\bar{a}_k$ if $\bar{a}_j \geq \bar{a}_k$ entrywise. Consider two such columns and let $\bar{x}^* = (x_1^*,\ldots,x_j^*,\ldots,x_k^*,\ldots,x_n^*)$ solve (8.1). Then $(x_1^*,\ldots,x_j^* + x_k^*,\ldots,0_k,\ldots,x_n^*)$ is feasible to (8.1) and, moreover, has the same (optimal) objective value. Therefore

<u>Lemma 8.1:</u> An optimal solution to problem (8.1) exists for which none of the columns of A corresponding to nonzero variables are dominated by any other such column of A.

Therefore we may reduce (8.1) by eliminating any columns of A (and associated variables) which are dominated. But then the resulting matrix displays properly compatible circular 1's in columns, so that the problem is solvable by the approach just presented. (Note that, in fact, it is sufficient for this conclusion to assume so-called "agreeable" costs, for which $c_j \leq c_k$ <u>iff</u> $\bar{a}_j \geq \bar{a}_k$ (cf., [14])).

Let us assume that the matrix A has been pruned of dominated columns. Then the special properties of the transformed problem are of interest. In particular, the new objective function is $(\bar{c}T)\bar{y} = \bar{e}_n\bar{y} = \bar{0}\bar{y}_r + y_n$. Thus solving the transformed problem (6.2) is tantamount to finding the smallest integer y_n for which the constraints of (6.3) have a feasible solution. Equivalently, we seek the smallest integer y_n for which the dual network flow problem to (6.3) is not unbounded, i.e., is free of cycles of positive net cost.

For the special objective function $\bar{c}\bar{x}$ such that $c_1 \geq c_2 \geq \cdots \geq c_{n-1}$ (which includes the objective function $\bar{1}\bar{x}$), a particularly simple solution technique applies to the

transformed problem (6.3). The new objective function has the property $(\bar{c}T_r) \geq \bar{0}$; furthermore $\begin{bmatrix} A_r \\ T_r \end{bmatrix}$ has no more than one +1 in each row, and at least one +1 in each column. Thus this version of (6.3) is solvable by the simple recursive substitution scheme of Dorsey, Hodgson, and Ratliff [7].

9. Close Enough

For the transformed version of problem (8.1), an interesting round-off result holds (see similar results in [4,20]). Recall that the transformed, equivalent version of (8.1) is

$$\min \bar{0}\bar{y}_r + y_n$$

$$\text{s.t.} \begin{bmatrix} AT_r & \bar{a}_n \\ T_r & \bar{e}_n \end{bmatrix} \begin{bmatrix} \bar{y}_r \\ y_n \end{bmatrix} \geq \begin{bmatrix} \bar{b} \\ \bar{0} \end{bmatrix} \tag{9.1}$$

$$\bar{y}_r,\ y_n \text{ unrestricted but integer}$$

<u>Lemma 9.1</u>: Let $\bar{y}' = (y_1', \ldots, y_n')$ solve the continuous-valued relaxation of (9.1) then $\bar{y}^* = (\lceil y_1' \rceil, \lceil y_2' \rceil, \ldots, \lceil y_n' \rceil)$ solves the integer-restricted problem (9.1).

<u>Proof</u>: Clearly $\lceil y_n' \rceil$ is a lower bound on the optimal function value of (9.1). Moreover $(\lceil y_1' \rceil, \lceil y_2' \rceil, \ldots, \lceil y_n' \rceil)$ is an integer-valued vector which achieves this value. To see that this vector is feasible to (9.1), we will show that it satisfies each of the constraints, of which there are three types:

(i) $y_j - y_k \geq b_i$

(ii) $y_j - y_k + y_n \geq b_i$

(iii) $y_j \geq b_i$

First observe that for any two numbers a and b,

$$-\lceil a \rceil = \lfloor -a \rfloor, \tag{9.2}$$

and

$$\lceil a \rceil + \lceil b \rceil \geq \lceil a+b \rceil \tag{9.3}$$

By (9.3), $\lceil a-b \rceil + \lceil b \rceil \geq \lceil a \rceil$ so that $\lceil b \rceil - \lceil a \rceil \geq -\lceil a-b \rceil = \lfloor b-a \rfloor$ by (9.2). Then by the last inequality we have

(i) $\lceil y_j' \rceil - \lceil y_k' \rceil \geq \lfloor y_j'-y_k' \rfloor \geq \lfloor b_i \rfloor = b_i$ since b_i is integer.

(ii) $\lceil y_j' \rceil - \lceil y_k' \rceil + \lceil y_n' \rceil \geq \lceil y_j'+y_n' \rceil - \lceil y_k' \rceil \geq$ $\lfloor y_j'-y_k'+y_n' \rfloor \geq \lfloor b_i \rfloor = b_i$ since b_i integer.

(iii) $\lceil y_j' \rceil \geq y_j' \geq b_i$.

Hence, each of the constraints of (9.1) is satisfied and $(\lceil y_1' \rceil, \lceil y_2' \rceil, \ldots, \lceil y_n' \rceil)$ is an optimal feasible solution.

Q.E.D.

Therefore problem (8.1) may be solved by the following simple application of linear programming:

(i) Solve the continuous-valued relaxation of (8.1) by, for example, the simplex method of linear programming. Let the solution be $\bar{x}'$.

(ii) Transform the solution via $\bar{y}' = T^{-1}\bar{x}'$, for T as in (1.3).

(iii) Round-up $\bar{y}^* = (\lceil y_1' \rceil, \lceil y_2' \rceil, \ldots, \lceil y_n' \rceil)$.

(iv) Transform back to $\bar{x}^* = T\bar{y}^*$. Then $\bar{x}^*$ solves the integer program (8.1).

10. Applications

A. Cyclic Staffing with Overtime

A basic staffing problem involves a facility such as a hospital that operates 24 hours each day. Assume there are fixed hourly staff requirements b_i, and that there are three basic work shifts, each of eight hours duration: 0700-1500, 1500-2300, and 2300-0700. Overtime of up to an additional eight hours is possible for each shift. What is the minimum cost number of workers and their shifts such that all staff requirements are met? This problem may be formulated as in Figure 10.1, where the constraint matrix displays properly compatible circular 1's in columns. Thus the problem is efficiently solvable by a bounded series of network flow problem.

B. Days-off Scheduling

A problem studied by Brownell and Lowerre [5] is to minimize the total workforce necessary to meet daily staffing requirements, where each worker is guaranteed two days off each week, including every other weekend. For the case in which the days off each week are to be consecutive, the problem may be formulated as in Figure 10.2. The rows of the matrix display more complicated cyclic structure than simple circular 1's; but since the matrix has circular 1's in <u>rows</u>, the same change of variables transforms the problem to efficiently solvable form.

min $\overline{cx}$

s.t.

$$
\begin{array}{c}
07\\ 08\\ 09\\ 10\\ 11\\ 12\\ 13\\ 14\\ 15\\ 16\\ 17\\ 18\\ 19\\ 20\\ 21\\ 22\\ 23\\ 24\\ 01\\ 02\\ 03\\ 04\\ 05\\ 06
\end{array}
\begin{bmatrix}
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1
\end{bmatrix}
\bar{x} \geq \bar{b}
$$

$\bar{x} \geq \bar{0}$, integer

Figure 10.1: A cyclic staffing problem with overtime.

$$\min\ \bar{c}_1\bar{x}_1 + \bar{c}_2\bar{x}_2$$

$$\text{s.t.}\quad \left[\begin{array}{cccccc|cccccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
\hline
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right] \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \geq \bar{b}$$

$$\bar{x}_1,\ \bar{x}_2 \geq \bar{0}, \text{ integer}$$

Figure 10.2: A version of the Brownell and Lowerre problem.

REFERENCES

[1] Aho, A. V., J. E. Hopcroft, J. D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley Publishing Company, Reading, Massachusetts, 1974.

[2] Baker, K. R., "Scheduling a Full-Time Work Force to Meet Cyclic Staffing Requirements," Management Science Vol. 20, No. 12, August 1974.

[3] Baker, K. R., "Workforce Allocation in Cyclical Scheduling Problems: Models and Applications," Paper No. 122, June 1975, Graduate School of Business Administration, Duke University, Durham, N. C.

[4] Bartholdi, J. J. and H. D. Ratliff, "Unnetworks, with Applications to Idle Time Scheduling," Research Report No. 77-4, April 1977, Industrial and Systems Engineering Department, University of Florida, Gainesville, Florida.

[5] Brownell, W. W. and J. M. Lowerre, "Scheduling of Workforce Required in Continuous Operations Under Alternative Labor Policies," Management Science, Vol. 22, No. 5, January 1976, pp. 597-605.

[6] Chen, D. S., "A Simple Manpower Scheduling Algorithm for Two Consecutive Days Off," presented at the ORSA/TIMS Joint National Meeting, Miami, Florida, November 1976.

[7] Dorsey, R. C., T. J. Hodgson, H. D. Ratliff, "An Efficient Integer Programming Algorithm for a Multi-Facility, Multi-Production Production Scheduling Problem," Research Report No. 73-8, January 1973, Industrial and Systems Engineering Department, University of Florida, Gainesville, Florida.

[8] Edmonds, J. and R. M. Karp., "Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems," Journal for the Association of Computing Machinery, Vol. 19, No. 2, April 1972, pp. 248-64.

[9] Geoffrion, A. and R. Nauss, "Parametric and Post-optimality Analysis in Integer Linear Programming," Management Science, Vol. 23, No. 5, January 1977, pp. 453-66.

[10] Guha, D., "An Optimal Procedure for Allocating Man-power with Cyclic Requirements: General Case," Working Paper, February 1976, courtesy of D. Guha, The Port Authority of New York and New Jersey, One Path Plaza, Third Floor, Jersey City, N. J. 07306.

[11] Iri, M., "A Criterion for the Reducibility of a Linear Programming Problem to a Linear Network Flow Problem," RAAG Research Notes, Third Series, No. 98, February 1966.

[12] Iri, M., "On the Synthesis of Loop and Cutset Matrices and the Related Problems," RAAG Memoirs, Vol. 4, 1968, pp. 4-38.

[13] Koehler, G., "$\gamma_{b,n}$ Matrices," Working Paper, August 1976, Department of Management, University of Florida, Gainesville, Florida.

[14] Lawler, E., "Sequencing the Weighted Number of Tardy Jobs," Rev. Francaise Automat. Informat. Recherche Operationelle, to appear.

[15] Orlin, J., "Quick Optimal Weekly Scheduling with Two Consecutive Days Off," Technical Report 77-1, January 1977, Department of Operations Research, Stanford University, Stanford, California.

[16] Padberg, M. W., "Characterizations of Totally Uni-modular, Balanced, and Perfect Matrices," in B. Roy (ed.), Combinatorial Programming: Methods and Applications, B. Reidel Publishing Co., Dordrecht, Holland, 1975.

[17] Tibrewala, R., D. Philippe, and J. Browne, "Optimal Scheduling of Two Consecutive Idle Periods," Management Science, Vol. 19, No. 1, September 1972, pp. 71-5.

[18] Tucker, A., "Matrix Characterizations of Ciruclar Arc Graphs," Pacific Journal of Mathematics, Vol. 39, No. 2, 1971, pp. 535-45.

[19] Veinott, A. F., Jr. and H. M. Wagner, "Optimal Capacity Scheduling - I and II," Operations Research, Vol. 10, No. 4, 1962, pp. 518-46.

[20] Weinberger, D. B., "Network Flows, Minimum Coverings, and the Four Color Conjecture," Operations Research, Vol. 24, No. 2, March-April 1976, pp. 272-90.

[21] Wilde, D. J., Optimum Seeking Methods, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1964.